

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Absolvování individuální odborné praxe
Individual Professional Practice in the Company

2011

Lukáš Bednárek

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární
prameny a publikace, ze kterých jsem čerpal.

V Ostravě 6. Května 2011

.....

Rád bych na tomhle místě poděkoval všem zaměstnancům firmy EG – Expert s.r.o. bez kterých by tato práce nikdy nemohla vzniknout.

Abstrakt

Moje bakalářská práce se vztahuje k vykonané praxi u firmy EG – Expert s.r.o. Moje práce ve firmě představovala pomoc při realizaci projektu pro firmu DisTep. Mohl jsem si vyzkoušet mnoho moderních technologií vývoje databázových aplikací.

Klíčová slova

Programovací jazyk SQL, programovací jazyk PL/SQL, programovací jazyk C#, relační datový model, aplikace eSADA, subsystém Balance distribuce

Abstract

My baccalaureate work relates to the practice carried out at the firm EG - Expert Ltd. My job at the company conceived to help with the project for the company DisTep. I could try a lot of modern technology development of database applications.

Keywords

SQL programming language, PL/SQL programming language, C# programming language, relational data model, application eSADA, Balance distribution subsystem

Seznam použitých zkratk a symbolů

C++	-	programovací jazyk C++
C#	-	programovací jazyk C#
DCL	-	data control language
DDL	-	data definition language
DML	-	data manipulation language
ERD	-	entity relationship diagrams
IO	-	integritní omezení
IT	-	informační technologie
MS	-	Microsoft
PDA	-	personal digital assistant
PL/SQL	-	Procedural Language/Structured Query Language
SQL	-	Structured Query Language
SW	-	software
TCL	-	transaction control language
TID	-	Technologická informační databanka
XML	-	Extensible Markup Language

Obsah

1 Úvod	3
2 Popis odborného zaměření firmy	4
3 Zadané úkoly v průběhu odborné praxe.....	5
4 Zvolený postup řešení zadaných úkolů.....	6
4.1 Seznámení se s koncepcí a strukturou eSADA.....	6
4.2 Vytvoření datového modelu řešení	8
4.3 Logika subsystému Bilance distribuce.....	9
4.4 Zavádění objektů do systému eSADA	10
4.5 Migrace historických dat	11
4.6 Zpětná kontrola namigrovaných dat pomocí vizuální prezentace	11
5 Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe	12
5.1 Relační datový model	12
5.2 Jazyk SQL.....	12
5.3 Jazyk PL/SQL	13
5.4 Jazyk C#.....	13
6 Závěr	14
6.1 Znalosti či dovednosti scházející studentovi v průběhu odborné praxe.....	14
6.2 Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení.....	14
7 Reference	15

Seznam obrázků

Výrobní informační systém	6
Servisní informační systém.....	7
Zákaznický informační systém	8
Schéma logiky bilance distribuce	10
Struktura bloku PL/SQL	13

1 Úvod

Pro svou bakalářskou práci jsem se rozhodl využít možnosti absolvování individuální odborné praxe v mnou zvolené firmě. Praxi jsem absolvoval ve firmě EG – Expert s.r.o. ve Frýdku – Místku. Během praxe jsem se podílel na realizaci informačního systému pro firmu DisTep. Plnil jsem nejrůznější úkoly z oblasti vývoje tohoto systému, nejčastěji jsem pracoval s programovacím jazykem PL/SQL. Díky praxi jsem se seznámil prakticky s kompletním postupem při vytváření informačních systémů pro větší firmy.

2 Popis odborného zaměření firmy

Odbornou individuální praxi jsem absolvoval ve firmě EG – Expert, s.r.o. Firma byla založena v roce 1997 a její hlavní sídlo je v Trutnově, dále má pobočky v Praze, Frýdku – Místku a Jindřichově Hradci. Firma působí dlouhodobě v oblasti informačních technologií. Zaměřuje se hlavně na oblast pomocných obslužných programů, zejména pro energetiku, teplárenství a vodohospodářství.

Posláním firmy je poskytovat služby v oblasti IT efektivně, garantovaně a zároveň zvyšovat odbornost, jakost a komplexnost práce pro zákazníky v roli systémového integrátora. V současnosti má firma okolo 30 zaměstnanců a několik externích spolupracovníků.

Hlavní činnost firmy se zaměřuje na provozně-technické informační systémy určené zejména jako pomocné obslužné programy pro energetiku. EG – Expert je výrobcem a dodavatelem SW řešení eSADA, které se skládá ze tří částí:

- Výrobní informační systém (VIS) – určený pro plánování a bilancování výroby
- Servisní informační systém (SIS) – určený pro řízení údržby
- Zákaznický informační systém (ZIS) – určený pro faktORIZaci tepla, vody, plynu a elektřiny

Firma je také certifikovaným partnerem firmy Microsoft a provádí implementaci i ekonomických systémů na bázi produktu Microsoft Dynamics. Pro své zákazníky firma poskytuje dlouhodobý servis SW aplikací a metodickou pomoc. Nejznámějšími zákazníky firmy jsou například firma ČEZ a. s., ČEPS a. s., Dalkia ČR a. s. a mnoho teplárenských společností.

3 Zadané úkoly v průběhu odborné praxe

Během praxe jsem pomáhal při realizaci mnoha úkolů, některých složitějších a některých lehčích. Popíši zde jen ty stěžejní, které byli hlavní náplní mé práce.

Prvním úkolem, po příchodu na praxi, bylo seznámení se základním konceptem a strukturou celého řešení eSADA, což je provozně – technický informační systém určený pro energetiku. Tento systém činí základ celé mé práce.

Dalším zajímavým úkolem bylo vytvoření datového modelu řešení se zaměřením na agendu Evidence objektů a jejich výrobních charakteristik. Jedná se o zpracování podkladů od zákazníka a jejich převedení do datového modelu pro databázové schéma.

Jeden těžší úkol bylo naprogramovat v jazyce PL/SQL logiku subsystému Balance distribuce, kde jsem se měl hlavně zaměřit na předávací stanice. Šlo o různé výpočty, kontrolu vstupních dat apod. Nejspíše nejlehčím úkolem bylo vkládání objektů do systému eSADA, jelikož data od zákazníka se mění jen v určitých hodnotách, je jednodušší tato data vložit ručně pomocí připravených formulářů. Nejspíše nejsložitějším úkolem byla migrace historických dat. Jednalo se o import dat od zákazníka, podle kterých se následně aktualizovaly staré hodnoty za nové.

Posledním větším úkolem byla zpětná kontrola namigrovaných dat, šlo o kontrolu vložených dat, jestli mají správný tvar, jejich úplnost a následné vyhodnocení pomocí grafů, jestli odpovídají předpokladům pro různé roční období.

4 Zvolený postup řešení zadaných úkolů

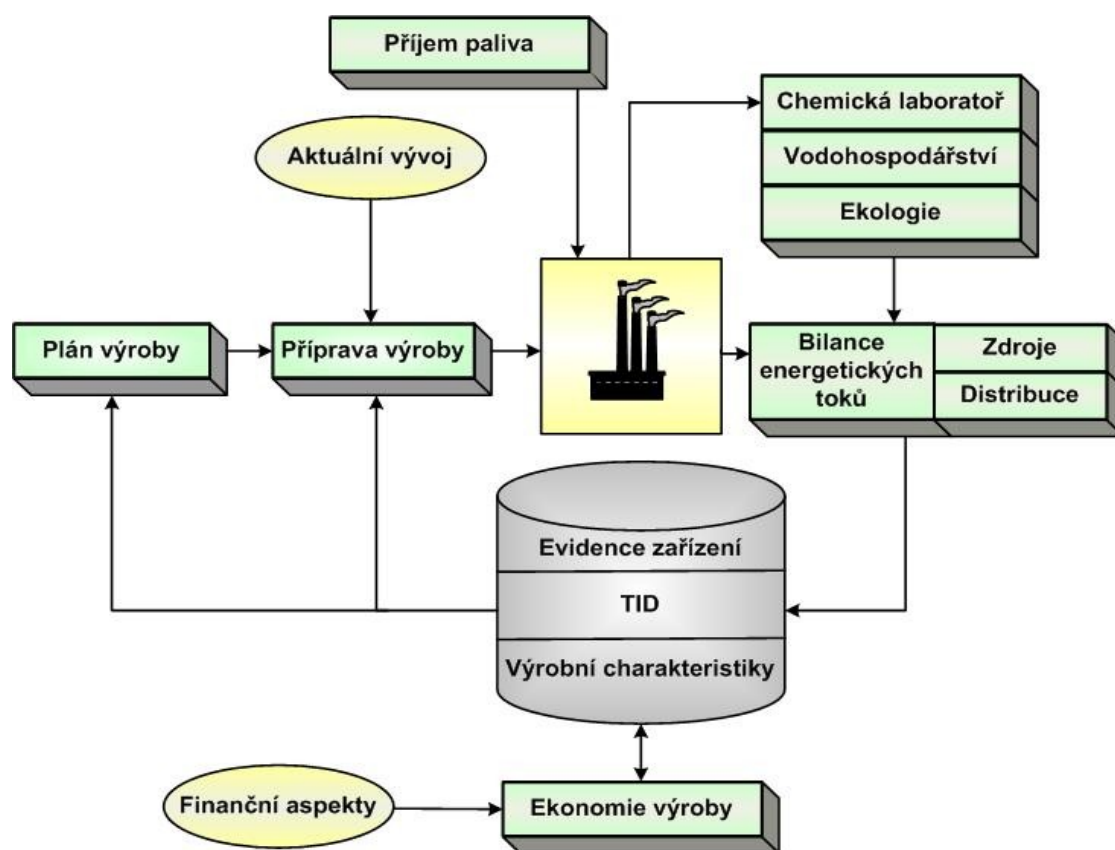
4.1 Seznámení se s koncepcí a strukturou eSADA

Po příchodu do firmy jsem dostal za úkol v první řadě se seznámit s aplikací eSADA, která je vyvíjena touto firmou a z ní vychází všechny jejich projekty pro různé energetické firmy. Tento systém obsahuje tři subsystémy, a také jeden databázový server, nejčastěji Oracle 11g. Tyto subsystémy jsou navzájem propojeny a tvoří jeden celistvý celek.

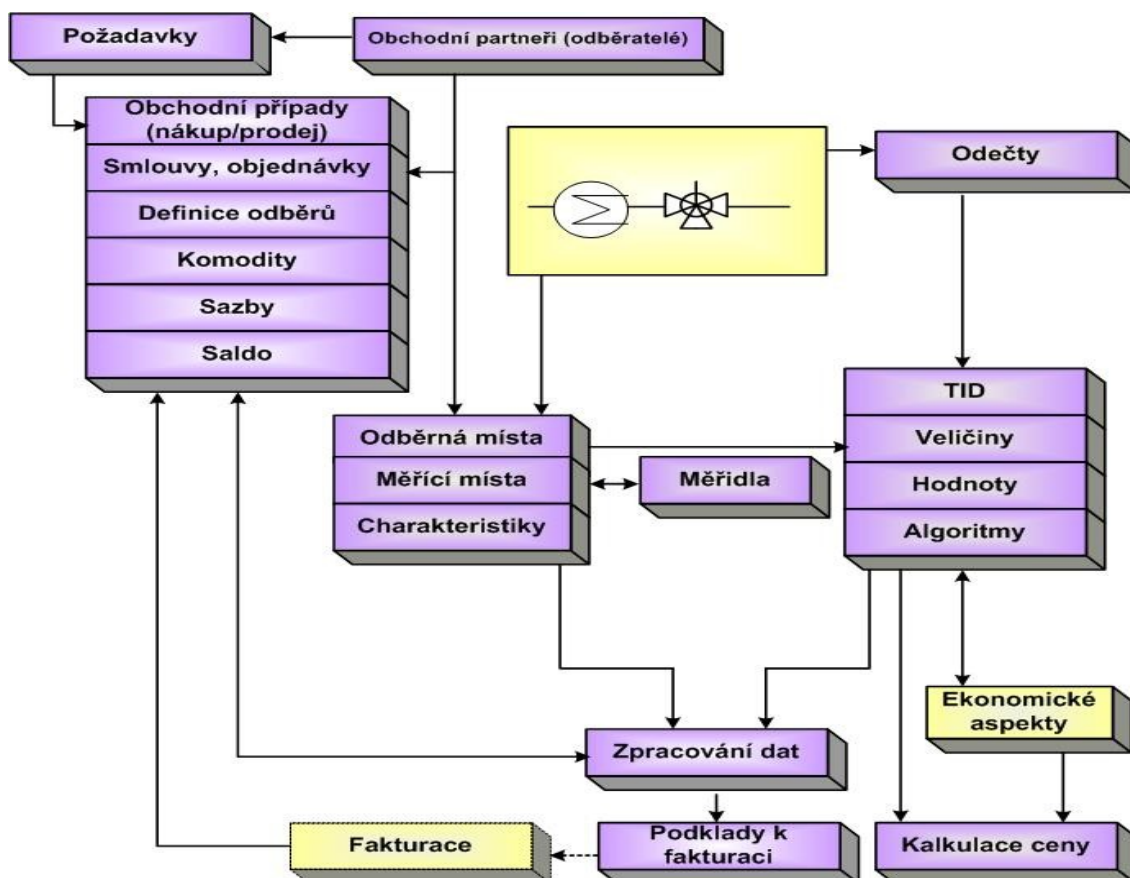
Prvním systémem je Výrobní informační systém (obr. 1), který obstarává řízení výroby a distribuci energetických komodit. Tento systém eviduje, co a kde bylo vyrobeno.

Dalším systémem je Servisní informační systém (obr. 2), který zajišťuje plynulý chod zařízení a řízení údržby těchto zařízení. Jedná se o evidenci zaměstnanců, informace o provedené práci apod.

Posledním systémem je Zákaznický informační systém (obr. 3). Tento systém obsluhuje zákazníka, tedy eviduje spotřebu různých veličin, dále tuto spotřebu propočítává a srovnává s plánovanou spotřebou. Taktéž provádí fakturaci spotřeby.



Obrázek 1: Výrobní informační systém



Obrázek 3: Zákaznický informační systém

4.2 Vytvoření datového modelu řešení

Pro tento úkol jsem se musel nejdříve seznámit s podklady a požadavky od firmy DisTep. Prakticky šlo o prvotní vytvoření konceptuálního schématu řešení a jeho následné převedení na datový model. Výsledek by měly být vlastně tři datové modely pro databázi, jelikož musíme oddělit různé části informačního systému od sebe. To znamená vytvořit zvlášť modely pro samotné objekty, zvlášť pro zákazníky a jako poslední část samotnou spotřebu a její fakturaci.

Tyto datové modely jsem postupně vytvářel pomocí programu TOAD Data Modeler [5], kde jsem definoval jména tabulek a jim příslušných atributů za pomoci dokumentace od zákazníka. Dále jsem definoval vazby mezi jednotlivými tabulkami a jejich kardinalitu, toto vše jsem průběžně konzultoval s ostatními pracovníky firmy, jelikož tento projekt je hlavně týmová práce. Po vytvoření jednotlivých logických datových modelů následovala opětovná důkladná kontrola, která měla za úkol odhalit případné nesrovnalosti, ovšem můj návrh byl správný. Poté probíhalo převedení logického datového modelu na fyzický, který již reprezentuje všechny tabulky, atributy a jejich vzájemné vazby v databázi. Nakonec jsem z výsledného fyzického modelu vygeneroval SQL skripty, které se následně implementovaly do vyvíjeného informačního systému.

4.3 Logika subsystému Bilance distribuce

Logikou subsystému Bilance výroby a distribuce se rozumí vytvoření prostředí pro výpočty a kontrolu dat vložených do systému a jejich zobrazení v různých časových intervalech. Hlavními úkoly bylo naprogramovat procedury a funkce v jazyce PL/SQL, dále pak vytvoření propočtů na jistý časový interval a také naimplementování tiskové sestavy, která bude schopna vytisknout požadovaná data s jejich konkrétními výpočty, a také možnost exportu výsledků do programu MS Excel.

Tento úkol byl celý implementován na straně databázového serveru, aby nedocházelo k zbytečně velkému přenosu dat v síti. Projekt byl realizován celý nad databázovým serverem Oracle 11g.

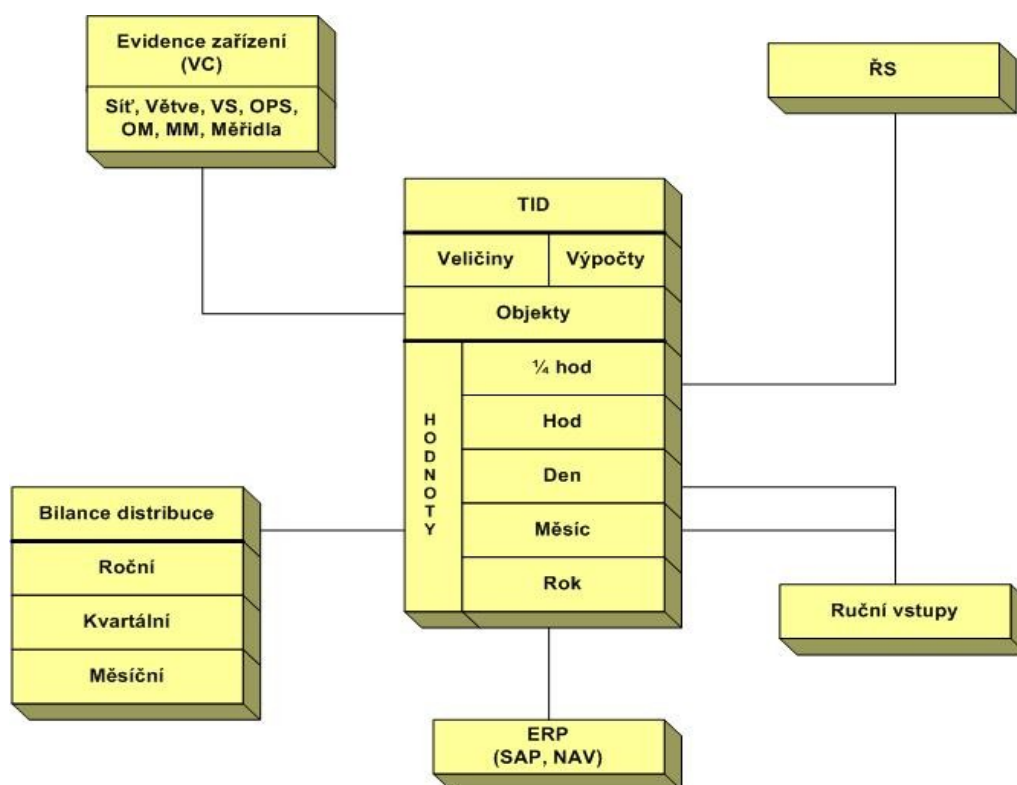
Pomocí programu SQL Developer jsem vytvářel postupně za pomoci firemních programátorů požadované procedury a funkce. Schéma celé logiky subsystému uvádím na obrázku 4.

Cílem bylo vytvořit uložené procedury pro výpočet celkové spotřeby, nebo také pro výpočet aktuální spotřeby v zadaném časovém intervalu, dále pak výpočet ztrát způsobených distribucí tepla pomocí venkovních vedení, výpočet celkových nákladů apod. Nejdříve jsem však musel naimplementovat funkce, které zaprvé zkontrolují požadovaná data a tím pádem zajistí jejich korektnost ve výpočtu. Hlavně bylo zapotřebí zkontrolovat úplnost dat, dále pak jejich jednotky, jestli odpovídají jednotkám používaným ve výpočtech a případně tato data upravit podle požadavků procedury.

Další částí tohoto úkolu bylo naimplementování převodu výsledků mezi časovými intervaly, to znamená například sečíst hodinové výsledky a z nich zobrazit součet a průměr za jeden den, nebo sečíst údaje za dny a vypočíst měsíční průměr apod.

Pro tiskovou sestavu se opět jednalo o uloženou proceduru, která měla zobrazit základní hodnoty veličin a jejich spotřebu za určité období, například spotřeba tepla za den na uvedeném místě.

Poslední část tohoto úkolu byl export všech vypočtených dat do formátu MS Excel, pro tento úkol měla firma již vytvořenou funkci, jelikož tato funkce je využívána pro více informačních systémů, bylo zapotřebí menších úprav pro tento systém. Tyto úpravy jsem s pomocí ostatních pracovníků úspěšně provedl a export dat fungoval správně.



Obrázek 4: Schéma logiky bilance distribuce

4.4 Zavádění objektů do systému eSADA

Tento úkol byl nejlehčí v celé praxi, ovšem neméně důležitý. Jelikož každý objekt v reálném světě má specifické hodnoty, je potřeba jej ručně vložit do aplikace eSADA. Data jsem vkládal do databázových tabulek, které jsem vytvořil na začátku praxe. Vkládání probíhalo přes formuláře v grafickém uživatelském rozhraní, které firemní programátoři vytvořili již dříve.

Mým úkolem byla evidence předávacích míst a k nim nadefinování měřících míst včetně vazeb na již osazená měřící místa a měřidla. Příkladem může být distribuce tepla ve městě. Z teplárny proudí teplo do předávací stanice, z ní následně do měřícího místa (například pata obytné budovy) a samotné jednotlivé měřidla (například jednotlivé ústřední topení).

4.5 Migrace historických dat

Migrací historických dat se rozumí naměřená absolutní spotřeba jednotlivých měřících míst za konkrétní uplynulé období.

Firma DisTep měla požadavek, aby historické údaje nebyly do systému vkládány jako koncové stavy měřidel, ale již jako spotřeba v daném časovém intervalu a daných technických jednotkách.

Tento úkol jsem realizoval s pomocí firemních programátorů, jelikož šlo o obsáhlejší řešení pomocí programovacího jazyka C#.

Zprv má program za úkol načíst data poskytnutá zákazníkem nejčastěji z formátu MS Excel. Po načtení je opět nutná kontrola a korekce dat. Dalším úkolem programu je převedení nestrukturovaných podkladů a jejich následné uspořádání do systémově správné a jednotné podoby. Například převedení veličin a jejich jednotek na základní míru apod.

Po kontrole a validaci všech vstupních dat jsem implementoval logiku pro správnou aktualizaci a přepočet spotřeby pro konkrétní nové hodnoty. Program využíval připojení na databázi, díky kterému jsem měl k dispozici mnou vytvořené procedury a funkce. Tyto prostředky jsem použil pro samotné výpočty, kontrolu a vložení nových dat do databáze.

4.6 Zpětná kontrola namigrovaných dat pomocí vizuální prezentace

Posledním větším úkolem, na kterém jsem se podílel, byla implementace nástroje pro zpětnou kontrolu vypočtených dat a její vizuální prezentace.

Programování opět probíhalo v programovacím jazyce PL/SQL. Výsledkem měla být procedura, která data zpětně zkontroluje a výsledky zobrazí na časové ose.

Nejdříve se vypočtená data zkontrolovala, jestli jsou úplná. Tedy všechny výsledky musí obsahovat správnou jednotkou míru, pro každé měřící místo musí být vypočtená spotřeba apod. Po úspěšné kontrole následovala jejich vizuální prezentace pomocí aplikace TID. Tato aplikace je naprogramována v jazyce C#. Je tedy nutné tuto aplikaci rozšířit o nástroj, který nám výsledné hodnoty spotřeby zobrazí v časovém grafu. Výsledný graf se zobrazí v aplikaci TID a posledním úkolem celé praxe je vizuální kontrola výsledných hodnot. Jedná se například o spotřebu tepla za určitý měsíc nebo za celý rok, kde se kontroluje hlavně průběh křivek. Například spotřeba tepla je v zimě mnohem větší než v létě a tudíž křivky by měly mít předpokládaný průběh. Jestliže se v grafu vyskytne nepředpokládaná hodnota, je nutná zpětná kontrola dat jejich výpočtů a celkově celého procesu migrace historických dat.

5 Teoretické a praktické znalosti a dovednosti získané v průběhu studia uplatněné studentem v průběhu odborné praxe

5.1 Relační datový model

Relační datový model[1] je základním prvkem uložení dat v databázi. Jedná se o uložení dat v logickém smyslu. Přístup k datům je symetrický, tj. nezajímá nás přístup k datům.

Základním stavebním prvkem tohoto modelu je tzv. relace (tabulka), která obsahuje n-tice (řádky). Tabulka je základ celé relační databáze. Každá tabulka je struktura záznamu s pevně stanovenými položkami (sloupci – atributy). Každý sloupec má dán název, typ a rozsah neboli doménu atributu. Pokud je ve více tabulkách sloupec stejného typu, pak tento sloupec můžeme použít pro vytvoření vazby mezi těmito tabulkami. Relační model klade velký význam na zachování integrity dat, zavádí tedy pojem referenční integrita, cizí klíč, primární klíč. S tímto modelem je úzce spjat jazyk SQL.

5.2 Jazyk SQL

Jazyk SQL je standardizovaný strukturovaný dotazovací jazyk používaný pro práci s daty v relačních databázích. Na začátku 70. let 20. století jej začala vyvíjet firma IBM, vývoj šel s vývojem relačních databází a bylo potřeba vytvořit sadu příkazů pro ovládání těchto databází. Cílem bylo vytvořit, ve kterém by se příkazy zadávali syntakticky co nejbližší k přirozenému jazyku, tedy angličtině.

Hlavními skupinami jazyka SQL jsou:

- Příkazy pro definici dat (DDL)
- Příkazy pro manipulaci s daty (DML)
- Příkazy pro řízení přístupu (DCL)
- Příkazy pro řízení transakcí (TCL)

První skupina má za úkol vytvořit, popřípadě upravit strukturu databáze, jako je například vytvoření tabulek. Typickými příkazy jsou CREATE, DROP, ALTER.

Do druhé skupiny patří příkazy, které jakkoliv manipulují s daty, například vypisují data podle různých kritérií, ukládání a mazání dat apod. Nejčastěji to jsou tato klíčová slova SELECT, INSERT, UPDATE, DELETE.

Skupina příkazů pro řízení přístupu má na starosti přidělovat nebo odebírat uživatelům práva na různé objekty databáze. Patří zde pouze příkazy GRANT a REVOKE.

Poslední skupinou jsou příkazy, které spravují databázové transakce, představiteli jsou BEGIN, COMMIT, ROLLBACK.

Příklady použití jazyka SQL naleznete zde [2].

5.3 Jazyk PL/SQL

Jazyk PL/SQL[3] je procedurální nadstavbou jazyka SQL. Je vyvinut firmou Oracle a jako základ posloužil programovací jazyk Ada. Tuto nadstavbu dnes již používají všechny velké databázové firmy.

Základním stavebním kamenem v PL/SQL je blok (Obrázek 5), jeden blok obvykle provádí jednu logickou akci v programu. Bloky se mohou do sebe vnořovat.

```
DECLARE
/* Deklarace obsahuje proměnné, typy a lokální subprogramy. */
BEGIN
/* Výkonná sekce: zde běží procedury a SQL kód. */
/* Toto je jediná sekce, která je v bloku povinná. */
EXCEPTION
/* Oblast spracování výjimek: zde se spracovávají chybové události. */
END;
```

Obrázek 5: Struktura bloku PL/SQL

Jedinou povinnou sekcí bloku je výkonná sekce, ve které můžeme použít pouze příkazy DML, poté pár příkazů pro řízení transakcí, příkazy DDL zde nejsou povoleny.

Základní prvky jazyka jsou různé proměnné, konstanty, atributy, kurzory. Dále lze v PL/SQL použít různé cykly, tak jako v jiných programovacích jazycích. Je zde také podpora výjimek, lze si definovat vlastní výjimky, to dělá tento jazyk mnohem mocnějším než jazyk SQL.

5.4 Jazyk C#

Jazyk C# [4] vyvinula firma Microsoft jako vysoce úroňový objektově orientovaný programovací jazyk, vývin se prováděl paralelně s platformou .NET Framework. Je založen na jazycích C++ a Java.

Používá se hlavně k výrobě webových služeb, od databází po různé formuláře ve Windows, taktéž pro výrobu software pro mobilní zařízení jako PDA.

Hlavní cíle tohoto jazyka je podpora principů pro softwarové inženýrství jako například hlídání hranic polí, automatický garbage collector atd. Jazyk je vhodný pro vývoj softwarových komponent v různých prostředích.

Hlavními vlastnostmi jazyka C# jsou:

- Podpora vývoje založeného na komponentách (metadata)
- Integrovaná XML dokumentace
- Delegát – bezpečné předávání funkcí jako parametrů
- Rozsáhlý systém knihovny tříd
- Jednotný systém výjimek

6 Závěr

6.1 Znalosti či dovednosti scházející studentovi v průběhu odborné praxe

Potřebné základní znalosti z oblastí návrhu datového modelu, programování v jazycích SQL, PL/SQL a C # mi byly poskytnuty školním studiem. Na praxi jsem si je ověřil a pomocí různých programátorských příruček i rozšířil své dosavadní znalosti a dovednosti.

6.2 Dosažené výsledky v průběhu odborné praxe a její celkové zhodnocení

Na praxi jsem se dozvěděl spoustu nových postupů a metod při programování v jazycích SQL, PL/SQL a C#. Byla to pro mě velice zajímavá zkušenost a určitě ji v budoucnu využiji.

7 Reference

- [1] Mgr. Kaluža, R. *Relační datový model*, [cit. 2011-3-18], <http://owebu.blogger.cz/Databaze/Relacni-datovy-model>
- [2] Bc. Hordějčuk, V. *Jazyk SQL*, [cit. 2011-3-18], <http://voho.cz/wiki/poznamky-sql/>
- [3] *PL/SQL*, [cit. 2011-3-18], <http://cs.wikipedia.org/wiki/PL/SQL>
- [4] *C#*, [cit. 2011-3-18], http://cs.wikipedia.org/wiki/C_Sharp
- [5] *TOAD Data Modeler*, [cit. 2011-3-18], <http://www.casestudio.com/enu/default.aspx>